

1

《C/C++ 学习 指南》

第09.1讲：指针的定义

作者：邵发 QQ群：417024631

官网：http://www.afanihao.cn/c_guide/

答疑：<http://www.afanihao.cn/kbase/>

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

2

版权所有，侵权必究

注意

提醒：你必须先前面的所有视频按顺序学完，否则听不懂这节课。

如果直接跳到本次课，则很可能听不懂。

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

背景知识：变量与内存

（回顾3.1，3.3讲的内容）

- (1) 内存用于存储数据，最小单元是字节（8bit），每个单元都有一个编号（即地址：
0x00000000~0xFFFFFFFF）
- (2) 变量有大小（sizeof操作符，int大小是4），变量的大小指的是它在内存中占了
几个字节。
- (3) 变量有地址，变量的地址就是该变量在内存中的位置。用&号可以取得地址
- (4) 变量赋值，就是向内存写入数据。读变量的值，就是从内存读取数据。
- (5) 每个变量都有自己的内存地址，绝不会有2个变量的内存地址相同。（一个萝卜
一个坑。。。）
- (6) 程序每次运行时，变量的地址是不一样的，无法预测也不需要预测其地址。

《C/C++学习指南》邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

查看变量的地址

地址是一个整数：0x00000000 ~ 0xFFFFFFFF

```
int a = 0;
double b = 0;
printf( "%08X \n", &a);
printf( "%08X \n", &b);
```

也可以

```
printf( "%p \n", &a); // p: pointer
```

《C/C++学习指南》邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

5

版权所有，侵权必究

如何表示地址？

地址是整数，是否可以用 int ？（理论上可以。。。）
但是为了强调它是个内存地址，提出一些新的类型。。。

char* : 表示一个char型变量的地址
short* : 表示一个short型变量的地址
int* : 表示一个int型变量的地址
float* : 表示一个float型变量的地址
double* : 表示一个double型变量的地址
unsigned char* : 表示一个unsigned char型变量的地址
...
XXX* : 表示XXX型变量的地址

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

6

如何表示地址？

以下以int型变量为例：

```
int a = 10;
int* pa = &a;
```

定义一个变量pa，类型为int*，初始值为&a。

有两种叫法：

- (1) pa指向了一个int型变量
- (2) pa所在的内存是一个int型整数

我们称int*为**指针类型**，pa为指针类型的变量（简称**指针**）

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

关于指针

(1) 指针也是变量，是可以变的。

```
int a = 10;
int b = 11;
int* p = &a; // p: 指向a
p = &b; // p: 指向b
```

可以把它视为一种整型变量。

(注意：这里只说了支持赋值操作符=，而关于加减乘除等其他操作符，后面再说。。。)

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

关于指针

(2) 不同类型的指针，不能互相赋值

```
int a = 10;
int* pa = &a; // pa: int*型
double* pb = pa; // 错！！ 左侧double*, 右侧int*
(int*: 指向int型变量, double*: 指向double型变量, 肯定不能互相赋值)
```

又如：

```
char a = 78;
float* p = &a; //错：左侧为float*, 右侧为char*
```

关于指针

(3) 指针即地址，地址是整数，所以指针是一个整数类型

```
int a ;  
int* p = &a;  
printf( "addr: %08X \n", p);
```

也可以使用 “%p”，效果和%X差不多

```
printf( "addr: %p\n", p);
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

关于指针

(4) 星号*的位置

以下几种均可以，没有区别，比较自由

```
int* a; // int与*连写  
int * a; // int与*中间有空白  
int *a; // *与变量名连着
```

都可以，哪种都差不多。作者的习惯是第一种写法。

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

星号操作：按地址访问

已知一个int型变量，有两种方式来改变它的值。

(1) 直接使用变量名赋值

```
int a = 0;
a = 0x11;
a = 0x12;
a = 0x1314;
```

(2) 使用指针(按地址访问，直接修改内存)

```
int a = 0;
int* p = &a; // p指向a
*p = 0x11;
*p = 0x1314;
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

星号操作：按地址访问

有一个指针变量p，则*p用于访问p指向的变量(p指向的内存)。

或者称作：***p用于访问p指向的内存(读、写)**

```
int a = 0x1314;
int* p = &a; // p指向a所在的内存
*p += 2;
```

```
int b = *p; // 取得p指向内存的值
int c = *p + 2;
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

星号操作：更多练习

是区分： 指针定义语句中的* 和 按地址访问的*

```
int a = 10;
```

```
int b = 11;
```

```
int* p ; // 定义一个指针，注意int*作为一个整体
```

```
p = &a; // p指向a
```

```
printf( " %d \n" , *p); // *p: 取得p处内存的值
```

```
p = &b; // p指向b
```

```
printf( " %d \n" , *p); // *p: 取得p处内存的值
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

星号操作：int不支持星号操作

虽然说指针是一种整数类型，但是它是特殊的，*号只能作用于指针类型。

```
int addr = 0x12345678; // addr: 普通int型变量
```

```
*addr = 0; // 编译错误！ 只有指针才能支持星号操作
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

星号操作：其他指针类型

其他指针类型的用法也是完全一样的。。。

```
char a = 78;  
char* p = &a;  
*p = 79;
```

```
double m = 1.1;  
double* pm = &m;  
*pm = 1.2
```

... 完全一样

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库

小结

(1) 提出新的类型 `char*` /`short*`/`int*`... 用于表示某型变量的地址。

```
int* p = &a; // p指向变量a, 或p指向a的内存
```

(2) 学习按地址访问：星号操作

```
*p = 1234;  
int b = *p;
```

《C/C++学习指南》 邵发 <http://afanihao.cn> 全套免费教学视频/配套书本/配套题库